

PATENT
2060-3-45
Customer No: 035884

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:
Sang Keun Lee
Serial No:
Filed: Herewith
For: SYSTEM AND METHOD FOR MAINTAINING
TRANSACTION CACHE CONSISTENCY IN MOBILE
COMPUTING ENVIRONMENT

Art Unit:

Examiner:

TRANSMITTAL OF PRIORITY DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Enclosed herewith is a certified copy of Korean patent application No. 10-2002-67237 which was filed on October 31, 2002 and from which priority is claimed under 35 U.S.C. Section 119 and Rule 55.

Acknowledgment of the priority document(s) is respectfully requested to ensure that the subject information appears on the printed patent.

Respectfully submitted,

Date: July 1, 2003

By: 

Jonathan Y. Kang
Registration No. 38,199
F. Jason Far-Hadian
Registration No. 42,523
Amit Sheth
Registration No. 50,176
Attorney for Applicant(s)

LEE & HONG
801 S. Figueroa Street, 14th Floor
Los Angeles, California 90017
Telephone: (213) 623-2221
Facsimile: (213) 623-2211



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Intellectual
Property Office.

출원 번호 : 10-2002-0067237
Application Number PATENT-2002-0067237

출원 년 월 일 : 2002년 10월 31일
Date of Application OCT 31, 2002

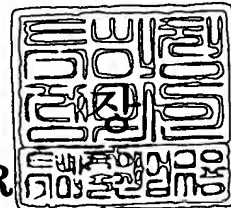
출원인 : 엘지전자 주식회사
Applicant(s) LG Electronics Inc.



2002 년 11 월 25 일

특 허 청

COMMISSIONER



【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0011
【제출일자】	2002. 10. 31
【국제특허분류】	H04B 13/00
【발명의 명칭】	이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법
【발명의 영문명칭】	SYSTEM AND METHOD FOR MAINTAINING CONSISTENCY OF TRANSACTION CACHE IN MOBILE COMPUTING ENVIRONMENTS
【출원인】	
【명칭】	엘지전자 주식회사
【출원인코드】	1-2002-012840-3
【대리인】	
【성명】	박장원
【대리인코드】	9-1998-000202-3
【포괄위임등록번호】	2002-027075-8
【발명자】	
【성명의 국문표기】	이상근
【성명의 영문표기】	LEE, Sang Keun
【주민등록번호】	711105-1697115
【우편번호】	140-031
【주소】	서울특별시 용산구 이촌1동 우성아파트 101동 1006호
【국적】	KR
【공개형태】	학술단체 서면발표
【공개일자】	2002.08.01
【심사청구】	청구
【취지】	특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인 박장원 (인)
【수수료】	
【기본출원료】	20 면 29,000 원
【가산출원료】	8 면 8,000 원

【우선권주장료】	0	건	0	원
【심사청구료】	8	항	365,000	원
【합계】	402,000		원	
【첨부서류】	1. 요약서·명세서(도면)_1통 2. 공지에외적용대상(신규성상 실의예외, 출원시의특례)규정을 적용받 기 위한 증명서류_1 통			

【요약서】**【요약】**

본 발명은 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법에 관한 것으로, 이러한 본 발명은 갱신된 데이터와 갱신 타임 스탬프를 포함하는 무효화 보고 메시지를 이동 서버에서 주기적으로 이동 클라이언트로 방송하고, 이동 클라이언트에 무효화 보고 메시지가 도착될 때마다 상기 도착된 무효화 보고 메시지를 이용하여 트랜잭션 캐시의 무효화 처리를 수행하며, 무효화 처리 결과를 바탕으로 캐시 일관성 유지 처리를 수행하고, 일관성이 유지된 캐시의 데이터들을 바탕으로 이동 클라이언트 독자적으로 읽기 전용 이동 트랜잭션의 완료 처리를 수행할 수 있도록 한다. 따라서 본 발명은 이동 서버의 부하를 이동 클라이언트로 분산하고, 이동 서버와 이동 클라이언트 간 송수신 메시지를 줄여 제한된 대역폭의 사용 효율을 높일 수 있다.

【대표도】

도 2

【색인어】

이동컴퓨팅, 읽기전용이동트랜잭션, 타임스탬프기반낙관적동시성제어, 트랜잭션캐시일관성 유지, 트랜잭션직렬성, 트랜잭션완료검사, 무효화보고방송

【명세서】**【발명의 명칭】**

이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법{SYSTEM AND METHOD FOR MAINTAINING CONSISTENCY OF TRANSACTION CACHE IN MOBILE COMPUTING ENVIRONMENTS}

【도면의 간단한 설명】

도1은 일반적인 이동 컴퓨팅 망의 구성을 보인 도면.

도2는 본 발명에 의한 이동 서버와 이동 클라이언트의 소프트웨어 블럭 구성을 보인 도면.

도3은 본 발명에 의한 이동 클라이언트의 읽기 전용 이동 트랜잭션 실행 방법을 보이는 도면.

도4a 및 도4b는 본 발명에 의한 이동 클라이언트의 트랜잭션 캐시 무효화 및 일관성 검사 수행 방법을 보이는 도면.

도5는 본 발명에 의한 이동 클라이언트의 읽기 전용 이동 트랜잭션 완료 처리 방법을 보이는 도면.

도면의 주요 부분에 대한 부호의 설명

10: 이동 클라이언트 11: 트랜잭션 실행 프로세스

12: 캐시 무효화 프로세스 13: 캐시 일관성 검사/유지 프로세스

14: 완료 검사/처리 프로세스

【발명의 상세한 설명】**【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

- <10> 본 발명은 이동 컴퓨팅 시스템에 관한 것으로, 특히 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법에 관한 것이다.
- <11> 무선 통신 기술의 발전과 이동 가능한 통신 기기의 대중화로 인하여 위치나 장소에 상관없이 사용자들이 이동 중에도 여러 가지 정보를 제공받을 수 있는 이동 컴퓨팅 환경이 대두되었다.
- <12> 이동 컴퓨팅 환경에서 트랜잭션은 이동 단말기를 통한 사용자의 주문이나 계산서 발행 작업 등과 같이 일괄처리를 필요로 하는 경우, 일부분의 서버 작업이 실패되면 전체 작업이 처리되지 않아야 하는 경우에 사용된다. 즉, 트랜잭션은 일부분의 서버 작업들이 전체적으로 모두 성공될 때 성공적으로 실행되는 작업에 사용되는 처리 기능이다. 따라서 하나의 트랜잭션은 하나 이상의 연산 단위로 이루어진다.
- <13> 이러한 이동 컴퓨팅 환경에서의 트랜잭션의 처리는 호스트의 이동성, 잦은 단절, 제한된 대역폭(bandwidth)과 배터리 용량 등에 제약을 받는다. 따라서 이동 컴퓨팅 환경에서 가능한 한 제한된 대역폭의 이용을 줄이고 이동 트랜잭션의 응답 시간을 향상시키기 위하여 자주 액세스되는 데이터를 이동 호스트가 캐싱하는 캐싱 기법이 제안되었다.
- <14> 도1은 이동 컴퓨팅 망의 일반적인 구성의 일례를 보인다.
- <15> 이동 컴퓨팅 망은 크게, 고정망(유선망)(30)과 유선으로 연결되어 이동하지 않는 고정 호스트와; 이동하는 이동 호스트;의 두가지 요소로 구성된다.

- <16> 여기서, 무선 통신 인터페이스를 가지며 서비스를 제공해주는 고정 호스트를 이동 서버(20)로 칭하고, 상기 이동 서버(20)와 무선 통신을 수행하는 이동 호스트를 이동 클라이언트(10)로 칭한다.
- <17> 하나의 이동 서버(20)가 서비스해줄 수 있는 영역을 셀(cell)이라 하며, 같은 셀 안에 위치해 있는 이동 클라이언트들(10)은 동일한 이동 서버(20)와 무선 채널을 통해 통신한다. 상기 셀은 cellular connection으로 이루어질 수도 있고 무선 LAN(Local Area Network)으로 이루어질 수도 있다.
- <18> 일반적으로 이동 서버(20)는 넓은 대역폭(bandwidth)과 풍부한 배터리를 가지고 있으며, 유선 망(30)을 이용하여 통신하므로 신뢰적인 통신을 수행할 수 있다. 그러나, 이동 클라이언트(10)는 제한된 대역폭과 배터리를 가지고 있으며, 무선 망을 이용하여 통신하므로 통신의 신뢰성이 떨어진다.
- <19> 이동 클라이언트(10)는 이동하거나 단절(disconnection)된 동안에도 캐싱된 데이터를 이용하여 트랜잭션을 수행할 수 있으며, 이동 클라이언트(10)로 제출되는 트랜잭션을 이동 트랜잭션이라 한다.
- <20> 이동 클라이언트(10)의 캐시에 캐싱된 데이터와 동일한 데이터가 이동 서버(20)에서 갱신되는 경우, 이동 클라이언트(10)의 캐시 일관성(consistency)을 유지하기 위해 이동 서버(20)는 이동 클라이언트(10)에게 캐싱된 데이터가 갱신되었음을 알려주는 무효화 보고(Invalidation Report; IR) 메시지를 발송한다.

- <21> 그러나 이동 컴퓨팅 망은 제한된 대역폭(limited bandwidth)을 가지고 있기 때문에 이동 클라이언트(10)와 이동 서버(20) 사이의 잦은 메시지 교환은 대역폭에 병목 현상(bottleneck)을 발생시키는 요인이 된다.
- <22> 그래서 이동 서버(20)는 이동 클라이언트(10)와 가능한 한 통신을 줄이면서 이동 클라이언트(10)에게 데이터의 갱신 여부를 알려 주는 것이 바람직하다. 따라서, 이동 서버(20)는 일정 주기 동안 갱신된 정보를 모아 두었다가 주기적으로 이동 클라이언트(10)에게 무효화 보고 메시지를 발송한다.
- <23> 이때, 이동 클라이언트(10)는 주기적으로 발송되는 무효화 메시지를 수신하기 전까지는 이동 서버(20)에서 발생한 데이터의 갱신 여부를 알지 못한다.
- <24> 이동 트랜잭션의 직렬성(serializability)이란, 한 개 이상의 트랜잭션이 동시에 수행된 결과는 순차적(직렬적)으로 실행된 결과와 동일하다는 특성을 의미한다.
- <25> 그런데, 상기와 같이, 캐싱된 데이터가 이동 서버(20)에서 갱신되는 경우 이동 클라이언트(10)가 무효화 메시지에 대한 발송을 수신할 때까지 데이터의 갱신 여부를 알지 못하므로, 이동 클라이언트(10)의 캐시 일관성이 보장될 수 없고 이동 트랜잭션의 직렬 가능한 수행이 보장될 수 없는 문제점이 발생되었다.
- <26> 또한 종래 이동 컴퓨팅 환경에서 이동 클라이언트가 읽기 전용 이동 트랜잭션을 실행한 후 이동 서버에게 상기 실행된 읽기 전용 이동 트랜잭션의 완료를 요청하면, 이동 서버는, 상기 완료 요청에 따라 상기 읽기 전용 이동 트랜잭션을 완료 또는 철회를 결정하고, 그 결정 사항을 이동 클라이언트에게 알려주었다. 그러면, 이동 클라이언트는 이동 서버로부터 전송되는 결정 사항에 따라 상기 읽기 전용 이동 트랜잭션을 완료 또는

철회하였다. 따라서 종래 이동 컴퓨팅 환경에서는 이동 서버와 이동 클라이언트 간에 많은 양의 송수신 메시지가 요구되며 이에 따라 제한된 대역폭의 효율적 사용을 방해하는 문제가 발생되었다.

<27> 그리고 이동 클라이언트에서 실행된 읽기 전용 이동 트랜잭션의 완료/철회 결정이 이동 서버에 의해 이루어지므로 긴 응답 시간이 요구되는 문제점도 있었다.

<28> 또한 이동 서버가 이동 클라이언트에서 실행된 이동 트랜잭션의 완료/철회의 결정 까지도 수행해야 하므로 많은 부하가 부여되는 문제점도 있었다.

【발명이 이루고자 하는 기술적 과제】

<29> 본 발명은 상기와 같은 종래 기술의 문제점을 해결하기 위해 제안된 것으로, 본 발명의 목적은 이동서버와 이동 클라이언트간 대역폭을 효율적으로 이용하고 이동 서버의 부하를 이동 클라이언트로 분산할 수 있도록 한 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.

<30> 본 발명의 다른 목적은 이동 컴퓨팅 환경에서 이동 클라이언트의 트랜잭션 캐시의 일관성을 용이하게 유지할 수 있도록 한 타임 스탬프 기반 낙관적 동시성 제어 (Optimistic Concurrency Control with Update TimeStamp Span; OCC-UTS²) 프로토콜을 제공함에 있다.

<31> 본 발명의 또다른 목적은 읽기 전용 이동 트랜잭션의 완료 처리를 이동 클라이언트에서 독자적으로 수행할 수 있도록 트랜잭션 캐시의 일관성을 유지하기 위한 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.

- <32> 본 발명의 또다른 목적은 이동 클라이언트가 타임 스탬프에 기반하여 트랜잭션 캐시 데이터의 일관성을 검사하고 트랜잭션 완료 처리를 독자적으로 수행함으로써, 이동 서버의 부하를 분산할 수 있고 또한 이동 서버와 이동 클라이언트 간 송수신 메시지를 최소화할 수 있는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.
- <33> 본 발명의 또다른 목적은 이동 서버가 하나의 트랜잭션이 갱신한 데이터들에 대해 동일한 갱신 타임 스탬프를 부여하고, 이동 클라이언트가 무효화된 데이터들을 사용하고 있는 트랜잭션들을 모두 철회함으로써, 여러 데이터들간에 일관성을 유지할 수 있는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.
- <34> 본 발명의 또다른 목적은 타임 스탬프에 기반하여 이동 트랜잭션 완료 처리를 수행함으로써 빠른 트랜잭션 응답 시간을 제공할 수 있는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.
- <35> 본 발명의 또다른 목적은 이동 클라이언트가 타임 스탬프에 기반하여 트랜잭션 캐시 데이터의 일관성을 검사함으로써 즉시 캐싱된 데이터에 대해서도 일관성을 유지할 수 있는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템 및 방법을 제공함에 있다.
- <36> 상기와 같은 목적을 달성하기 위하여 본 발명에 의한 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템은, 주기적으로 무효화 보고 메시지를 방송하는 이동 서버와; 상기 무효화 보고 메시지가 도착하면 상기 무효화 보고 메시지의 타임스탬프를 이용하여 트랜잭션 캐시의 무효화 처리를 수행하고 무효화 처리 결과를 바탕으로 트랜잭

션 캐시의 일관성 유지를 위한 처리를 수행하며 실행된 읽기 전용 이동 트랜잭션의 완료 처리를 수행하는 이동 클라이언트;를 포함하여 이루어진 것을 특징으로 한다.

- <37> 상기와 같은 목적을 달성하기 위하여 본 발명에 의한 이동 컴퓨팅 환경에서의 이동 클라이언트의 트랜잭션 캐시 일관성 유지 방법은, 읽기 전용 이동 트랜잭션을 실행하는 과정과; 무효화 보고 메시지가 주기적으로 도착할 때마다 상기 무효화 보고 메시지의 타임 스탬프를 이용하여 트랜잭션 캐시 무효화 및 읽기 연산중인 데이터의 일관성 검사를 수행하는 과정과; 상기 실행된 읽기 전용 이동 트랜잭션을 완료 처리하는 과정;을 포함하여 이루어진 것을 특징으로 한다.

【발명의 구성 및 작용】

- <38> 트랜잭션들은 동시에 수행될 수 있기 때문에, 동시에 일어나는 다수 개 트랜잭션들의 실행을 관리하는 동시성 제어가 필요하다. 낙관적인 방법(optimistic approach)은, 동시성 제어를 위한 일반적인 방법들 중 하나이며, 데이터베이스 연산의 대부분이 충돌을 일으키지 않는다는 가정에 바탕을 둔 방법으로서, 데이터베이스를 읽거나 질의하는 연산으로만 구성된 읽기 전용 이동 트랜잭션에 바람직하다. 또한 이동 클라이언트와 이동 서버 간에 요구되는 메시지 교환을 최소화할 수 있기 때문에, 본 발명은 낙관적 동시성 제어를 적용한다. 또한 본 발명은 트랜잭션 캐시 일관성 및 무효화 검사가 용이하고 이동 클라이언트가 독자적으로 트랜잭션 완료 처리를 수행할 수 있도록 타임 스탬프 기반 낙관적 동시성 제어를 제안한다.

- <39> 이하, 첨부한 도면을 참조하여 본 발명의 실시예를 설명하면 다음과 같다.

- <40> 도1에 도시된 이동 컴퓨팅 망을 이용하여 본 발명의 실시예를 설명한다.
- <41> 도2는 본 발명의 일 실시예에 의한 이동 컴퓨팅 환경에서의 트랜잭션 캐시 일관성 유지 시스템의 소프트웨어 블록 구성을 보인다.
- <42> 도2에 도시된 바와 같이, 트랜잭션 캐시 일관성 유지 시스템은, 주기적으로 무효화 보고를 방송하는 이동 서버(20)와; 읽기 전용 이동 트랜잭션을 실행하며, 상기 무효화 보고가 도착하면 캐시의 무효화 및 일관성을 검사하고, 독자적으로 상기 읽기 전용 이동 트랜잭션의 완료 처리를 수행하는 이동 클라이언트(10);를 포함하여 구성된다.
- <43> 상기 캐시는, (데이터, 그 데이터의 갱신 타임스탬프 또는 캐시 타임 스탬프)의 집합이다.
- <44> 상기 이동 서버(20)는, 주기적으로 무효화 보고 메시지(Invalidation Report message; IR message)를 방송하는 무효화 보고 프로세스(21)와; 상기 이동 클라이언트(10)로부터 즉시 캐싱 요청이 송신되면, 데이터 세트를 제공하는 데이터 세트 제공 프로세스(22);를 포함한다.
- <45> 상기 IR 메시지는, 방송 타임 스탬프(ts_i)를 가지며, 갱신된 데이터와 갱신 타임 스탬프를 쌍(pair)으로 여러 쌍 포함한다. 상기 방송 타임 스탬프(ts_i)는 IR 메시지가 방송된 시간을 나타내며($ts_i = iL$), 여기서 L은 브로드캐스트 주기를 나타낸다. 상기 갱신 타임 스탬프는, 그 데이터를 마지막으로 갱신한 트랜잭션의 완료 타임 스탬프(commit time stamp)를 나타낸다. 상기 IR 메시지는, 이동 클라이언트(10)에 있는 트랜잭션 캐시의 데이터 무효화 및 일관성을 검사할 때 사용된다.

- <46> 상기 데이터 세트는, 유효(valid) 데이터와 그 유효 데이터의 갱신 타임 스탬프를 포함한다.
- <47> 상기 이동 클라이언트(10)는, 읽기 전용 이동 트랜잭션을 실행하는 트랜잭션 실행 프로세스(11)와; 주기적으로 도착되는 상기 무효화 보고 메시지를 이용하여 트랜잭션 캐시의 무효화를 검사하는 캐시 무효화 프로세스(12)와; 상기 무효화 검사시 무효화된 데이터를 이용하고 있는 다른 트랜잭션들이 있으면, 그 트랜잭션들을 철회하여 트랜잭션 캐시 일관성을 유지하는 캐시 일관성 검사/유지 프로세스(13)와; 상기 트랜잭션 실행 프로세스(11)가 접근한 데이터들이 갖는 캐시 타임 스탬프를 이용하여 즉시 유효화 완료 처리와 지연 유효화 완료 처리를 수행하는 완료 검사/처리 프로세스(14);를 포함하여 구성된다.
- <48> 상기 즉시 유효화 완료 처리 방법은, 상기 트랜잭션 실행 프로세스(11)가 접근한 모든 데이터가 동일한 캐시 타임 스탬프를 가지면 즉시 읽기 전용 트랜잭션의 완료 처리를 수행하는 방법이다.
- <49> 상기 지연 유효화 완료 처리 방법은, 상기 트랜잭션 실행 프로세스(11)가 접근한 모든 데이터가 동일한 캐시 타임 스탬프를 가지지 않는다면, 다음 IR 메시지가 도착할 때까지 읽기 전용 이동 트랜잭션의 완료/철회 결정을 연기하는 처리 방법이다. 상기 다음 IR 메시지가 도착한 후, 읽기 세트에 있는 어떠한 데이터도 무효화되지 않고 캐시로부터 삭제되지 않으면, 이동 클라이언트(10)는 읽기 전용 이동 트랜잭션을 완료할 수 있다.
- <50> 상기 읽기 세트는, (읽기 전용 이동 트랜잭션에 의해 읽기 연산이 수행되고 있는 데이터, 그 데이터의 캐시 타임 스탬프)의 집합이다.

- <51> 상기와 같이 구성된 본 발명의 실시예에 의한 이동 컴퓨팅 환경에서의 트랜잭션 캐시 일관성 유지 시스템의 동작을 첨부된 도면을 참조하여 설명하면 다음과 같다.
- <52> 이동 서버(20)는, 하나의 트랜잭션이 여러 데이터를 갱신했을 때 해당 데이터 항목(data item)에 동일한 갱신 타임 스탬프를 부여한다. 이는 이동 클라이언트(10)가 트랜잭션 완료 검사시 모든 접근된 데이터가 동일한 갱신 타임 스탬프를 가졌을 때 안전하게 즉시 완료할 수 있는 근거가 된다.
- <53> 또한, 이동 서버(20)는, 갱신된 데이터에 대한 정보를 이동 클라이언트(10)에게 알려 주기 위해 주기적으로 IR 메시지를 무선 통신 인터페이스를 통해 방송한다.
- <54> 이동 클라이언트(10)는, 읽기 전용 이동 트랜잭션이 제출되면, 트랜잭션 캐시에 캐싱되어 있는 데이터를 인출하여 그 읽기 전용 이동 트랜잭션을 실행한다. 이와 동시에, 필요한 시점, 즉, IR 메시지가 도착되는 시점에 이동 클라이언트(10)는 상기 도착된 IR 메시지를 기초로 트랜잭션 캐시의 무효화 및 일관성을 검사한다. 그런 다음, 이동 클라이언트(10)는 독자적으로 상기 읽기 전용 이동 트랜잭션의 완료 처리를 수행한다.
- <55> 먼저, 읽기 전용 이동 트랜잭션이 실행되는 과정을 설명한다.
- <56> 도3은 이동 클라이언트(10)의 읽기 전용 이동 트랜잭션을 실행하는 방법을 보이는 흐름도이다.

- <57> 도3에 도시된 바와 같이, 읽기 전용 이동 트랜잭션이 이동 클라이언트(10)에 제출되면(S11), 트랜잭션 실행 프로세스(11)는, 상기 읽기 전용 이동 트랜잭션이 요구하는 데이터가 캐시(트랜잭션 캐시)에 존재하는 지를 확인한다(S12).
- <58> 상기 요구되는 데이터가 캐시에 존재하면, 트랜잭션 실행 프로세스(11)는, 캐시로부터 즉시 상기 요구되는 데이터를 인출하여 상기 읽기 전용 이동 트랜잭션에게 제공한다(S13).
- <59> 그러나, 상기 요구되는 데이터가 캐시에 존재하지 않으면, 트랜잭션 실행 프로세스(11)는, 상기 읽기 전용 이동 트랜잭션을 이동 서버(20)에게 송신하여, 상기 데이터를 요구한다. 그러면 이동 서버(20)의 데이터 세트 제공 프로세스(22)는, 해당 데이터 세트를 이동 클라이언트(10)에게 송신한다. 상기 해당 데이터 세트는, 상기 요구되는 데이터(j)와, 그 데이터의 갱신 타임 스탬프(t_j)를 포함한다.
- <60> 이동 클라이언트(10)는, 이동 서버(20)로부터 전신된 상기 데이터 세트를 수신하고(S14), 수신된 데이터 세트의 데이터를 캐시에 로딩하며 그 데이터의 캐시 타임 스탬프를 적절히 변경한다(S15-S17).
- <61> 즉, 이동 클라이언트(10)의 트랜잭션 실행 프로세스(11)는, 상기 데이터 세트의 갱신 타임 스탬프(t_j)와 마지막 IR 타임스탬프(마지막으로 수신한 IR의 방송 타임 스탬프; ts_{lb})를 비교한다(S15).
- <62> 상기 데이터 세트의 갱신 타임 스탬프(t_j)가 마지막 IR 타임스탬프(ts_{lb})보다 작으면, 트랜잭션 실행 프로세스(11)는, 캐시에 로딩된 상기 데이터에 대한 캐시 타임 스탬프(t^c_j)를 상기 마지막 IR 타임스탬프(ts_{lb})로 할당(assign)한다(S16).

- <63> 그러나, 데이터 세트의 갱신 타임 스탬프(t_j)가 마지막 IR 타임스탬프(ts_{lb})보다 작지 않으면, 트랜잭션 실행 프로세스(11)는, 상기 데이터의 캐시 타임 스탬프(t^c_j)를 상기 갱신 타임 스탬프(t_j)로 할당한다(S17).
- <64> 이렇게 이동 서버(20)로부터 즉시 캐싱된 데이터의 캐시 타임 스탬프를 트랜잭션 캐시의 일관성을 유지할 수 있도록 변경한 후, 이동 클라이언트(10)는 즉시 캐싱된 데이터를 읽기 전용 이동 트랜잭션에게 제공한다(S18).
- <65> 이동 클라이언트(10)가 상기와 같이 읽기 전용 이동 트랜잭션의 실행 단계를 처리하고 있는 동안, 이동 서버(20)가 주기적으로 방송하는 IR 메시지가, 이동 클라이언트(10)에 도착하면, 이동 클라이언트(10)는 IR 메시지를 이용하여 캐시 무효화 처리와 캐시의 데이터 일관성 검사를 수행한다. 따라서, 주기적인 IR 메시지의 도착에 따라, 이동 클라이언트(10)는 캐시 무효화 처리와 캐시 데이터 일관성 검사를 주기적으로 수행하게 된다. 이동 클라이언트(10)의 캐시 무효화 처리와 캐시 데이터 일관성 검사에 대한 동작은 나중에 다시 상세히 설명하겠다.
- <66> 상기와 같이 읽기 전용 이동 트랜잭션이 실행되면, 이동 클라이언트(10)는 읽기 전용 이동 트랜잭션의 완료 처리를 독자적으로 수행한다.
- <67> 도5는 이동 클라이언트(10)가 읽기 전용 이동 트랜잭션의 완료 처리를 수행하는 방법을 보이는 흐름도이다.

- <68> 이동 클라이언트(10)의 트랜잭션 실행 프로세스(11)는 읽기 전용 이동 트랜잭션을 실행하고 완료 요청 메시지(commit request message)를 생성한다.
- <69> 이동 클라이언트(10)의 완료 검사/처리 프로세스(14)는, 상기 완료 요청 메시지를 수신하면(S41), 즉시 유효화 방법으로 상기 읽기 전용 이동 트랜잭션을 완료할 수 있는지를 체크한다(S42).
- <70> 즉, 상기 실행된 읽기 전용 이동 트랜잭션의 읽기 세트에 포함된 모든 데이터의 캐시 타임 스탬프가 모두 동일한 값을 가지는지를 체크한다(S43).
- <71> 상기 읽기 세트에 포함된 모든 데이터의 캐시 타임 스탬프가 모두 동일한 값을 가지면, 완료 검사/처리 프로세스(14)는 상기 실행된 읽기 전용 이동 트랜잭션을 즉시 완료한다(S44).
- <72> 그러나, 상기 읽기 세트에 포함된 모든 데이터의 캐시 타임 스탬프가 모두 동일한 값을 가지지 않으면, 완료 검사/처리 프로세스(14)는 다음 IR 메시지가 도착할 때까지 완료/철회의 결정을 연기한다(S45). 즉, 완료 검사/처리 프로세스(14)는 상기 읽기 전용 이동 트랜잭션의 완료를 요청하기 위한 완료 요청 메시지를 완료 요청 큐(Commit Queue)에 넣는다. 상기 완료 요청 메시지는, 완료 요청된 읽기 전용 이동 트랜잭션의 식별자(IDentity; ID)와 그 트랜잭션의 읽기 세트를 포함한다.
- <73> 이제, 이동 클라이언트(10)의 캐시 무효화 처리와 캐시 데이터 일관성 검사에 대한 동작을 설명한다.

- <74> 도4a 및 도4b는 이동 클라이언트(10)가 주기적으로 IR 메시지를 수신할 때마다 처리하는 동작을 보이는 흐름도이다.
- <75> 이동 서버(20)가 주기적으로 방송하는 IR 메시지가 이동 클라이언트(10)에 수신될 때마다(S21), 이동 클라이언트(10)의 캐시 무효화 프로세스(12)는, 도착된 IR 메시지($IR(ts_i)$)의 방송 타임스탬프(ts_i)와 이동 클라이언트(10)에 저장되어 있는 ts_{lb} 간의 차이가 L (브로드캐스트주기)보다 큰지를 확인한다(S22).
- <76> 상기 방송 타임 스탬프(ts_i)와 ts_{lb} 간의 차이가 L 보다 크면, 즉, 설정된 시간이 경과된 후 IR 메시지가 수신되었으면, 캐시 무효화 프로세스(12)는, 캐시에 저장되어 있는 모든 데이터를 삭제한다(S23).
- <77> 그러나, 상기 방송 타임 스탬프(ts_i)와 ts_{lb} 간의 차이가 L 보다 크지 않으면, 즉, 제 시간에 IR 메시지를 수신했으면, 캐시 무효화 프로세스(12)는, 상기 IR 메시지에 포함된 데이터와 동일한 데이터가 캐시에 있는 지를 확인한다(S24).
- <78> 상기 IR 메시지에 포함된 데이터와 동일한 데이터가 캐시에 없으면, 캐시 무효화 프로세스(12)는, 캐시에 있는 모든 데이터의 캐시 타임 스탬프를 방송 타임 스탬프로 할당한다(S28). 이렇게 본 발명은 무효화되지 않은 데이터에 대해서도 캐시 타임 스탬프를 갱신함으로써 읽기 전용 이동 트랜잭션의 완료 처리를 이동 클라이언트(10) 독자적으로 수행할 수 있게 된다.
- <79> 그러나, 상기 IR 메시지에 포함된 데이터와 동일한 데이터가 캐시에 있으면, 캐시 무효화 프로세스(12)는, 상기 IR 메시지에 포함된 데이터의 갱신 타임 스탬프(t_j)가 캐시에 있는 해당 데이터의 캐시 타임 스탬프(t_{cj})보다 큰 지를 확인한다(S25).

- <80> 상기 갱신 타임 스탬프(t_j)가 상기 캐시 타임 스탬프(t_{c_j})보다 크면, 캐시 무효화 프로세스(12)는, 상기 해당 데이터를 캐시로부터 삭제하여 무효화시킨다(S26). 그러나 상기 갱신 타임 스탬프(t_j)가 상기 캐시 타임 스탬프(t_{c_j})보다 크지 않으면, 캐시 무효화 프로세스(12)는, 상기 캐시 타임 스탬프(t_{c_j})를 상기 IR 메시지의 방송 타임스탬프(ts_i)로 할당한다(S27).
- <81> 캐시 무효화 프로세스(12)는 ts_{lb} 를 방송 타임스탬프(ts_i)로 할당한다(S29).
- <82> 그런 다음, 완료 검사/처리 프로세스(14)는, 완료 요청 큐(commit queue)가 비어 있는 지를 확인한다(S30). 완료 요청 큐가 비어 있지 않으면, 완료 검사/처리 프로세스(14)는, 완료 요청 큐로부터 완료 요청 메시지를 인출(dequeue)한다(S31).
- <83> 상기 인출된 완료 요청 메시지가 캐시 무효화 프로세스(12)에 의해 캐시로부터 삭제된 데이터를 포함하고 있으면, 완료 검사/처리 프로세스(14)는, 상기 완료 요청 메시지에 포함된 읽기 전용 이동 트랜잭션을 철회한다(S33). 그러나, 상기 인출된 완료 요청 메시지가 상기 삭제된 데이터를 포함하고 있지 않으면, 완료 검사/처리 프로세스(14)는, 상기 완료 요청 메시지에 포함된 읽기 전용 이동 트랜잭션을 완료한다(S34).
- <84> 이렇게 완료 검사/처리 프로세스(14)는 다음 IR 메시지를 수신한 후 캐시 무효화 프로세스(12)에 의해 캐시 무효화가 수행된 다음 캐시 무효화 처리 결과를 바탕으로 지연 유효화 완료 처리를 수행한다.
- <85> 이렇게 지연 유효화 완료 처리가 수행된 후, 캐시 일관성 검사/유지 프로세스(13)는, 캐시 무효화 프로세스(12)에 의해 캐시로부터 삭제된 데이터를 사용하고 있는 다른 트랜잭션들이 있는 지를 확인한다(S35).

<86> 캐시 일관성 검사/유지 프로세스(13)는, 상기 삭제된 데이터를 사용하고 있는 다른 트랜잭션들이 있으면, 해당 트랜잭션들을 철회한다(S36).

<87> 이와 같이, IR 메시지가 도착될 때마다, 이동 클라이언트(10)는, 캐시 무효화 처리와 캐시 데이터의 일관성을 유지하기 위한 처리를 주기적으로 수행한다.

【발명의 효과】

<88> 이상에서 살펴본 바와 같이, 본 발명에 의한 이동 컴퓨팅 환경에서의 트랜잭션 캐시 일관성 유지 시스템 및 방법은, 타임 스탬프 기반 낙관적 동시성 제어 (Optimistic Concurrency Control with Update TimeStamp Span; OCC-UTS²) 프로토콜을 읽기 전용 이동 트랜잭션에 적용하여, 읽기 전용 이동 트랜잭션의 직렬 가능한 수행을 보장하고 이동 클라이언트의 캐시 데이터의 일관성을 유지할 수 있으며, 이동 클라이언트 독자적으로 읽기 전용 이동 트랜잭션의 완료 처리를 수행할 수 있는 효과가 있다.

<89> 본 발명은 이동 클라이언트가 타임 스탬프에 기반하여 트랜잭션 캐시 데이터의 일관성을 검사하고 트랜잭션 완료 처리를 독자적으로 수행함으로써, 이동 서버의 부하를 분산할 수 있고 또한 이동 서버와 이동 클라이언트 간 송수신 메시지를 최소화할 수 있는 효과가 있다.

<90> 본 발명은 이동 클라이언트에 의한 읽기 전용 이동 트랜잭션 처리시 이동 서버와 이동 클라이언트 간에 동시성 제어를 위한 메시지 송수신을 완전히 제거함으로써 무선 인터페이스를 위한 대역폭을 효율적으로 사용할 수 있는 효과가 있다.

- <91> 본 발명은 이동 서버가 하나의 트랜잭션이 갱신한 데이터들에 대해 동일한 갱신 타임 스탬프를 부여하고, 이동 클라이언트가 무효화된 데이터들을 사용하고 있는 트랜잭션들을 모두 철회함으로써, 단일 데이터의 일관성이 아니라, 여러 데이터들간의 일관성을 유지할 수 있는 효과가 있다.
- <92> 본 발명은 이동 클라이언트가 타임 스탬프에 기반하여 이동 트랜잭션 완료 처리를 수행함으로써 빠른 트랜잭션 응답 시간을 제공할 수 있는 효과가 있다.
- <93> 본 발명은, 이동 클라이언트에 IR 메시지가 도착될 때마다, 무효화되지 않은 캐시의 데이터에 대해서도 상기 IR 메시지를 이용하여 캐시 타임 스탬프를 갱신함으로써 이동 클라이언트의 캐시 일관성을 유지할 수 있는 효과가 있다.
- <94> 본 발명은, 이동 서버가 데이터와 함께 타임스탬프를 제공하기 때문에, 이동 클라이언트가 데이터를 즉시 캐싱하는 경우에도 캐시 일관성을 유지할 수 있고 트랜잭션의 직렬 가능한 수행을 보장할 수 있는 효과가 있다.

【특허청구범위】**【청구항 1】**

주기적으로 무효화 보고 메시지를 방송하는 이동 서버와;

상기 무효화 보고 메시지가 도착하면 트랜잭션 캐시의 무효화 처리를 수행하고 무효화 처리 결과를 바탕으로 트랜잭션 캐시의 일관성 유지를 위한 처리를 수행하며 실행된 읽기 전용 이동 트랜잭션의 완료 처리를 수행하는 이동 클라이언트;를 포함하여 구성된 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템.

【청구항 2】

제 1 항에 있어서, 상기 이동 서버는,

하나의 트랜잭션이 다수 개의 데이터를 갱신했을 때 그 다수 개 데이터의 갱신 타임 스탬프를 모두 동일한 값으로 부여하는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템.

【청구항 3】

제 1 항에 있어서, 상기 이동 클라이언트는,

상기 실행된 읽기 전용 이동 트랜잭션의 완료가 요청되었을 때, 상기 읽기 전용 이동 트랜잭션의 읽기 세트에 포함된 모든 데이터들이 동일한 캐시 타임 스탬프를 가지면 즉시 완료 처리를 수행하고, 동일한 캐시 타임 스탬프를 가지지 않으면 다음 무효화 보고 메시지가 도착할 때까지 완료 처리를 지연시키는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 시스템.

【청구항 4】

읽기 전용 이동 트랜잭션을 실행하는 과정과;

무효화 보고 메시지가 주기적으로 도착할 때마다 트랜잭션 캐시 무효화 및 읽기 연산중인 데이터의 일관성 검사를 수행하는 과정과;

상기 실행된 읽기 전용 이동 트랜잭션을 완료 처리하는 과정;을 포함하여 이루어진 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 방법.

【청구항 5】

제 4 항에 있어서, 상기 캐시 무효화를 수행하는 과정은,

상기 무효화 보고 메시지가 브로드캐스트 주기 이내에 수신되지 않았으면, 트랜잭션 캐시의 모든 데이터를 삭제하는 과정과;

상기 무효화 보고 메시지가 브로드캐스트 주기 이내에 수신된 경우, 상기 무효화 보고 메시지에 포함된 데이터와 동일한 데이터가 트랜잭션 캐시에 존재하면, 그 데이터의 갱신 타임 스탬프와 캐시 타임 스탬프를 비교하는 과정과;

상기 갱신 타임 스탬프가 캐시 타임 스탬프보다 크면, 트랜잭션 캐시로부터 상기 데이터를 삭제하여 무효화시키는 과정과;

상기 갱신 타임 스탬프가 캐시 타임 스탬프보다 크지 않으면, 상기 캐시 타임 스탬프를 상기 무효화 보고 메시지의 방송 타임 스탬프로 할당하는 과정과;

상기 무효화 보고 메시지에 포함된 데이터와 동일한 데이터가 트랜잭션 캐시에 존재하지 않으면, 트랜잭션 캐시의 모든 데이터의 캐시 타임 스탬프를 상기 무효화 보고

메시지의 방송 타임 스탬프로 할당하는 과정;을 포함하는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 방법.

【청구항 6】

제 4 항에 있어서, 상기 데이터 일관성 검사를 수행하는 과정은,

상기 트랜잭션 캐시 무효화 처리에서 무효화된 데이터를 사용하고 있는 다른 트랜잭션들이 있으면, 그 트랜잭션들을 철회하는 과정;을 포함하는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 방법.

【청구항 7】

제 4 항에 있어서, 상기 완료 처리하는 과정은,

상기 읽기 전용 이동 트랜잭션이 실행된 후 완료 요청되면, 그 읽기 전용 이동 트랜잭션의 읽기 세트에 포함된 모든 데이터들이 동일한 캐시 타임 스탬프를 갖는 지를 확인하는 과정과;

상기 모든 데이터들이 동일한 캐시 타임 스탬프를 갖는다면, 즉시 상기 읽기 전용 이동 트랜잭션을 완료 처리하는 과정과;

상기 모든 데이터들이 동일한 캐시 타임 스탬프를 갖지 않는다면, 다음 무효화 보고 메시지가 도착할 때까지 상기 읽기 전용 이동 트랜잭션의 완료/철회 결정을 연기하는 과정;을 포함하는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 방법.

【청구항 8】

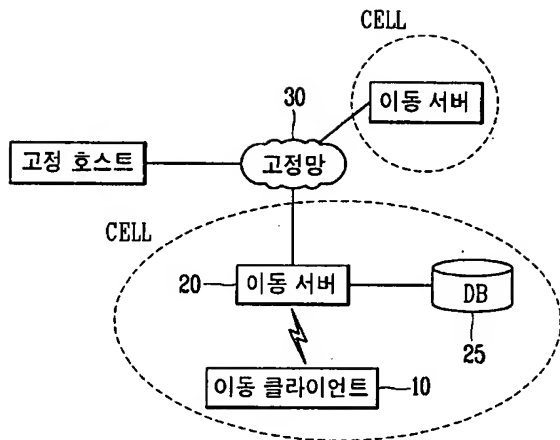
제 7 항에 있어서,

상기 다음 도착된 무효화 보고 메시지에 의해 트랜잭션 캐시로부터 무효화된 데이터를 상기 읽기 전용 트랜잭션이 포함하고 있지 않으면, 상기 읽기 전용 이동 트랜잭션을 완료하는 과정과;

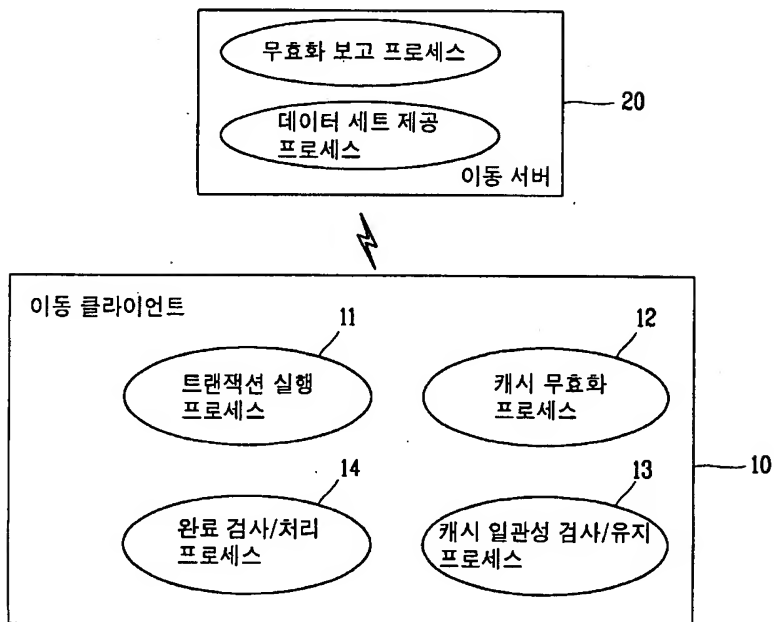
상기 무효화된 데이터를 상기 읽기 전용 트랜잭션이 포함하고 있으면, 상기 읽기 전용 이동 트랜잭션을 철회하는 과정;을 더 포함하는 것을 특징으로 하는 이동 컴퓨팅 환경에서 트랜잭션 캐시 일관성 유지 방법.

【도면】

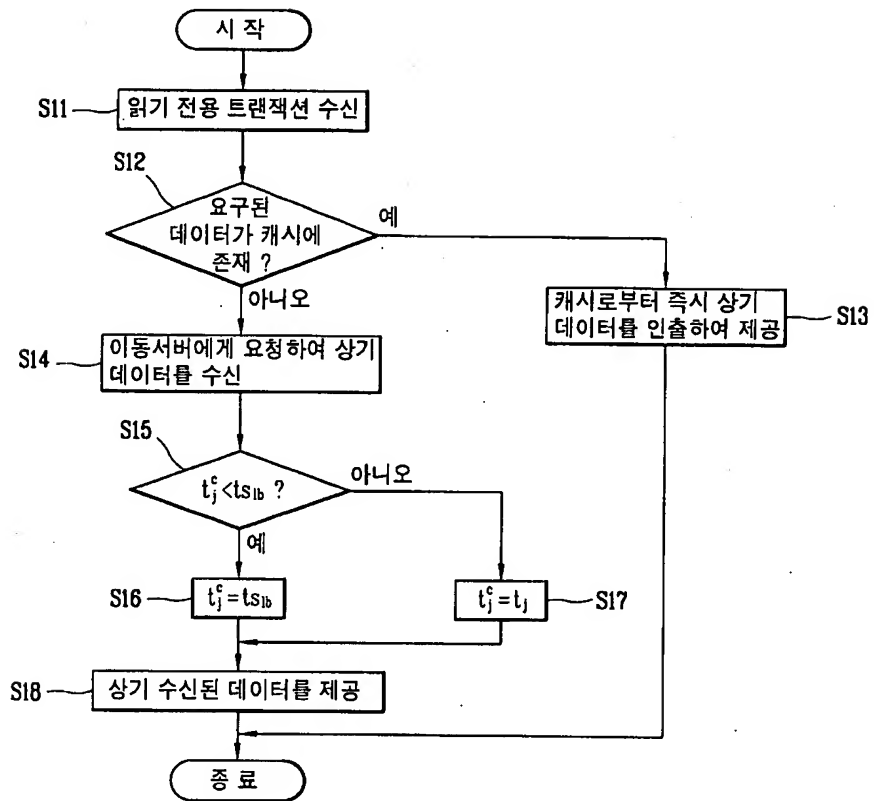
【도 1】



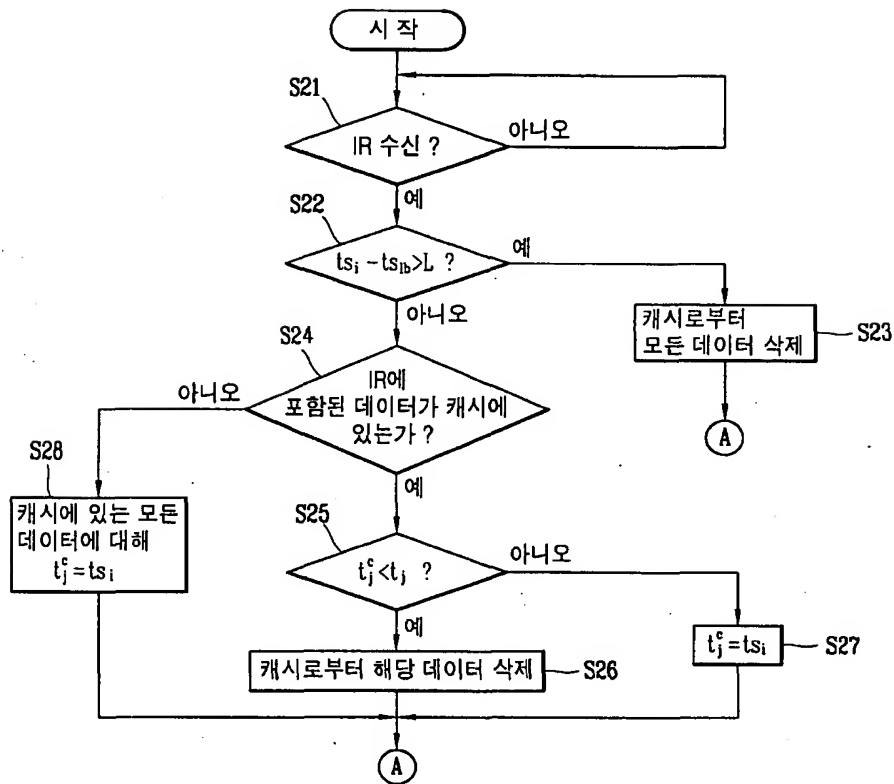
【도 2】



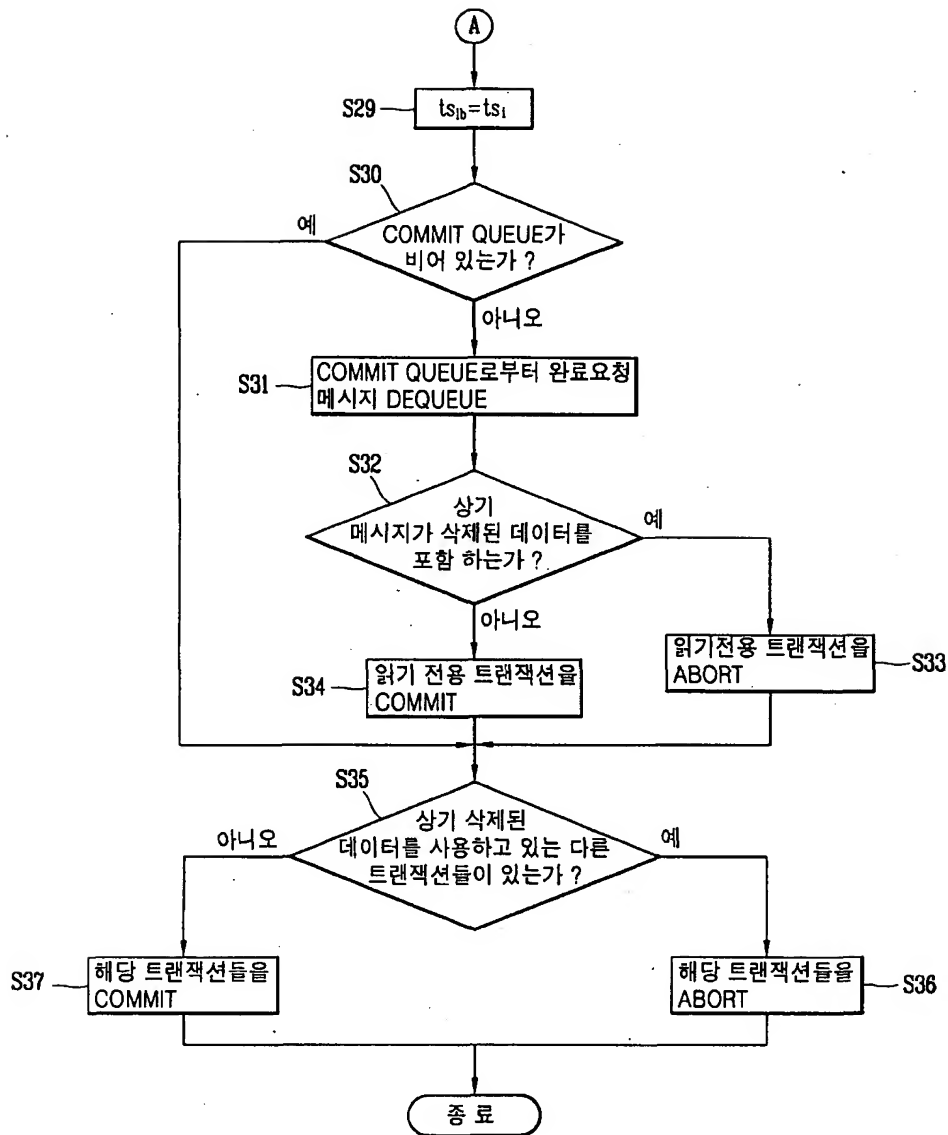
【도 3】



【도 4a】



【도 4b】



【도 5】

